UNITED STATES PATENT AND TRADEMARK OFFICE

UNITED STATES DEPARTMENT OF COMMERCE
United States Patent and Trademark Office
Address: COMMISSIONER FOR PATENTS
P.O. Box 1450
Alexandria, Virginia 22313-1450
www.uspto.gov

APPLICATION NO.	FILING DATE	FIRST NAMED INVENTOR	ATTORNEY DOCKET NO.	CONFIRMATION NO.
09/759,742	01/12/2001	Ralf Hofmann	P-4356	9744

24209	7590	10/16/2006

GUNNISON MCKAY & HODGSON, LLP
1900 GARDEN ROAD
SUITE 220
MONTEREY, CA 93940

EXAMINER
QUELER, ADAM M

ART UNIT	PAPER NUMBER
2178	

DATE MAILED: 10/16/2006

Please find below and/or attached an Office communication concerning this application or proceeding.

	Application No.	Applicant(s)
Office Action Summary	09/759,742	HOFMANN ET AL.
	Examiner	Art Unit	
	Adam M. Queler	2178	

-- *The MAILING DATE of this communication appears on the cover sheet with the correspondence address* --

Period for Reply

A SHORTENED STATUTORY PERIOD FOR REPLY IS SET TO EXPIRE <u>3</u> MONTH(S) FROM THE MAILING DATE OF THIS COMMUNICATION.

- Extensions of time may be available under the provisions of 37 CFR 1.136(a). In no event, however, may a reply be timely filed after SIX (6) MONTHS from the mailing date of this communication.
- If the period for reply specified above is less than thirty (30) days, a reply within the statutory minimum of thirty (30) days will be considered timely.
- If NO period for reply is specified above, the maximum statutory period will apply and will expire SIX (6) MONTHS from the mailing date of this communication.
- Failure to reply within the set or extended period for reply will, by statute, cause the application to become ABANDONED (35 U.S.C. § 133). Any reply received by the Office later than three months after the mailing date of this communication, even if timely filed, may reduce any earned patent term adjustment. See 37 CFR 1.704(b).

Status

1)☒ Responsive to communication(s) filed on <u>01 August 2006</u>.

2a)☒ This action is **FINAL**. 2b)☐ This action is non-final.

3)☐ Since this application is in condition for allowance except for formal matters, prosecution as to the merits is closed in accordance with the practice under *Ex parte Quayle*, 1935 C.D. 11, 453 O.G. 213.

Disposition of Claims

4)☒ Claim(s) *1,3,5,7-10,13 and 15-24* is/are pending in the application.

    4a) Of the above claim(s) _____ is/are withdrawn from consideration.

5)☐ Claim(s) _____ is/are allowed.

6)☒ Claim(s) *1,3,5,7-10,13 and 15-24* is/are rejected.

7)☐ Claim(s) _____ is/are objected to.

8)☐ Claim(s) _____ are subject to restriction and/or election requirement.

Application Papers

9)☐ The specification is objected to by the Examiner.

10)☐ The drawing(s) filed on _____ is/are: a)☐ accepted or b)☐ objected to by the Examiner.

    Applicant may not request that any objection to the drawing(s) be held in abeyance. See 37 CFR 1.85(a).

    Replacement drawing sheet(s) including the correction is required if the drawing(s) is objected to. See 37 CFR 1.121(d).

11)☐ The oath or declaration is objected to by the Examiner. Note the attached Office Action or form PTO-152.

Priority under 35 U.S.C. § 119

12)☐ Acknowledgment is made of a claim for foreign priority under 35 U.S.C. § 119(a)-(d) or (f).

    a)☐ All b)☐ Some * c)☐ None of:

      1.☐ Certified copies of the priority documents have been received.

      2.☐ Certified copies of the priority documents have been received in Application No. _____.

      3.☐ Copies of the certified copies of the priority documents have been received in this National Stage application from the International Bureau (PCT Rule 17.2(a)).

    * See the attached detailed Office action for a list of the certified copies not received.

Attachment(s)

1)☐ Notice of References Cited (PTO-892)

2)☐ Notice of Draftsperson's Patent Drawing Review (PTO-948)

3)☐ Information Disclosure Statement(s) (PTO-1449 or PTO/SB/08) Paper No(s)/Mail Date _____.

4)☐ Interview Summary (PTO-413) Paper No(s)/Mail Date. _____ .

5)☐ Notice of Informal Patent Application (PTO-152)

6)☐ Other: _____.

DETAILED ACTION

1.   This action is responsive to communications: Request for Reconsideration filed

08/01/2006

2.   Claims 1, 3, 5, 7-10, 13, 15-24 are pending in the case. Claims 1, 3, 8, 10, 13, 18, 20, 21,

and 24 are independent claims.

Claim Rejections - 35 USC § 103

3.   The following is a quotation of 35 U.S.C. 103(a) which forms the basis for all

obviousness rejections set forth in this Office action:

> (a) A patent may not be obtained though the invention is not identically disclosed or described as set forth in
> section 102 of this title, if the differences between the subject matter sought to be patented and the prior art are
> such that the subject matter as a whole would have been obvious at the time the invention was made to a person
> having ordinary skill in the art to which said subject matter pertains. Patentability shall not be negatived by the
> manner in which the invention was made.

4.   **Claims 1, 3, 5, 8-10, 13, 15, 16, 18, 19, 21 and 24 are rejected under 35 U.S.C. 103(a)**

as being unpatentable over Bayeh et al. (US006012098A patented 1/4/2000), and further in

view of Zeiger, "Servlet Essentials." "A History of Browsers" (hereinafter Quirksmode) is

cited as evidence regarding browsers.

Regarding independent claim(s) 1, 10, and 13, Bayeh teaches receiving a request for data (col.

4, ll. 23-29). Bayeh teaches retrieving a rule set for a plurality of filters, or servlets, (col. 9, ll.

46-63). The plain meaning of a filter registry, when viewed in light of the specification, is the

location of the rule set, which inherently the rule set must be retrieved from. Upon being

chained, the filters convert source data to requested data (col. 4, ll. 23-37). Bayeh teaches the

filter is a chain of partial filters (col. 9, ll. 30-46). Bayeh teaches that a subset of the data is

processed (col. 12, ll. 7-12). Bayeh does not explicitly disclose the exact manner in which the

servlets communicate with each other and explicitly suggests that one would look to the prior art

for guidance on such matters (col. 9, ll. 62-63 and col. 10, ll. 14-15). Zeiger discloses that a preferred way for servlets to communicate with other is to use a Java construct called an interface, which is used to ensure that different servlet types can communicate with each other (p. 30, para. 3). Applicant shows examples of interfaces (for example, Table 8 on page 29), but provides no clear definition of how "generic source and target data format independent" modifies the term "interface." Therefore, the term is treated as broadly as the plain meaning requires. As the interface is for servlets, which do not care about the underlying data format, merely that there is a text stream, the interface of Zeigler is not seen to be any different then the interface of the Applicant. Therefore it is a generic source and target data format independent interface, in accordance with the terms use in the specification. It would have been obvious to one of ordinary skill in the art at the time of the invention to use the interfaces of Ziegler to enable the intra-servlet communication necessary for Bayeh, because Bayeh suggests turning the prior art for implementation details, and because it would have enabled the servlets of Bayeh to be reloaded on the fly (p. 30, para. 1).

Regarding independent claim(s) 3, and 21, Bayeh teaches receiving a request for data (col. 4, ll. 23-29). Bayeh teaches a partial filter library as part of the server (col. 7, ll. 36-38). Bayeh teaches retrieving a rule set for a plurality of filters, or servlets, (col. 8, ll. 36-64), wherein upon being chained the filters convert source data to requested data (col. 4, ll. 23-37). Bayeh teaches the filter is a chain of partial filters, each of which as a generic format independent interface that passes data from one to another (col. 9, ll. 30-46). Bayeh teaches the filter converts source data from a first format to a second data format (col. 4, ll. 37-42).

Additionally, Bayeh teaches the client is a browser (col. 2, ll. 25-36). Bayeh does not

disclose a specific browser; therefore one of ordinary skill in the art would clearly turn toward an

accepted browser at the time of the invention. Quirksmode is cited as evidence that most

browsers at the time of the invention supported a plurality of formats, including at least HTML,

text (First Era, para. 1), images (First Era, para.1) and style sheets (Second Era, para. 6). It

would have been obvious to one of ordinary skill in the art at the time of the invention to turn to

the prior art for a browser as indicated by Bayeh.

Bayeh does not explicitly disclose the exact manner in which the servlets communicate with each

other and explicitly suggests that one would look to the prior art for guidance on such matters

(col. 9, ll. 62-63 and col. 10, ll. 14-15). Zeiger discloses that a preferred way for servlets to

communicate with other is to use a Java construct called an interface, which is used to ensure

that different servlet types can communicate with each other (p. 30, para. 3). Applicant shows

examples of interfaces (for example, Table 8 on page 29), but provides no clear definition of how

"generic source and target data format independent" modifies the term "interface." Therefore,

the term is treated as broadly as the plain meaning requires. As the interface is for servlets,

which do not care about the underlying data format, merely that there is a text stream, the

interface of Zeigler is not seen to be any different then the interface of the Applicant. Therefore

it is a generic source and target data format independent interface, in accordance with the terms

use in the specification. It would have been obvious to one of ordinary skill in the art at the time

of the invention to use the interfaces of Ziegler to enable the intra-servlet communication

necessary for Bayeh, because Bayeh suggests turning the prior art for implementation details,

and because it would have enabled the servlets of Bayeh to be reloaded on the fly (p. 30, para. 1).

Regarding independent claim(s) 24, Bayeh teaches receiving a request for data (col. 4, ll. 23-29). Bayeh teaches supporting different formats and selecting the second format (col. 8, ll. 55-57). Bayeh teaches generating a filter by combining a first filter with a second filter (col. 9, ll. 30-45). Bayeh teaches a using an XSL style sheet to transform the one format to the other. As each element must be mapped to the other format, this broadly encompasses a comparison of the two formats. Bayeh teaches the filter converts source data from a first format to a second data format (col. 4, ll. 37-42).

Bayeh does not explicitly disclose the exact manner in which the servlets communicate with each other and explicitly suggests that one would look to the prior art for guidance on such matters (col. 9, ll. 62-63 and col. 10, ll. 14-15). Zeiger discloses that a preferred way for servlets to communicate with other is to use a Java construct called an interface, which is used to ensure that different servlet types can communicate with each other (p. 30, para. 3). Applicant shows examples of interfaces (for example, Table 8 on page 29), but provides no clear definition of how "generic source and target data format independent" modifies the term "interface." Therefore, the term is treated as broadly as the plain meaning requires. As the interface is for servlets, which do not care about the underlying data format, merely that there is a text stream, the interface of Zeigler is not seen to be any different then the interface of the Applicant. Therefore it is a generic source and target data format independent interface, in accordance with the terms use in the specification. It would have been obvious to one of ordinary skill in the art at the time of the invention to use the interfaces of Ziegler to enable the intra-servlet communication necessary for Bayeh, because Bayeh suggests turning the prior art for implementation details, and because it would have enabled the servlets of Bayeh to be reloaded on the fly (p. 30, para. 1).

Regarding independent claim(s) 8 and 18, Bayeh teaches receiving a request for data (col. 4,

ll. 23-29). Bayeh teaches a partial filter library as part of the server (col. 7, ll. 36-38). Bayeh

teaches retrieving a rule set for a plurality of filters, or servlets, (col. 8, ll. 36-64), wherein upon

being chained the filters convert source data to requested data (col. 4, ll. 23-37). This chain of

filters is the general partial filter adapter as set forth in the claim language. Bayeh teaches the

filter is a chain of partial filters, each of which as a generic format independent interface that

passes data from one to another (col. 9, ll. 30-46). Bayeh teaches input to the filter is a XSL

style-sheet that determines its functionality (col. 9, ll. 4-6). As the filter is a servlet object, its

input is passed through a parameter.

Bayeh does not explicitly disclose the exact manner in which the servlets communicate

with each other and explicitly suggests that one would look to the prior art for guidance on such

matters (col. 9, ll. 62-63 and col. 10, ll. 14-15). Zeiger discloses that a preferred way for servlets

to communicate with other is to use a Java construct called an interface, which is used to ensure

that different servlet types can communicate with each other (p. 30, para. 3). Applicant shows

examples of interfaces (for example, Table 8 on page 29), but provides no clear definition of how

"generic source and target data format independent" modifies the term "interface." Therefore,

the term is treated as broadly as the plain meaning requires. As the interface is for servlets,

which do not care about the underlying data format, merely that there is a text stream, the

interface of Zeigler is not seen to be any different then the interface of the Applicant. Therefore

it is a generic source and target data format independent interface, in accordance with the terms

use in the specification. It would have been obvious to one of ordinary skill in the art at the time

of the invention to use the interfaces of Ziegler to enable the intra-servlet communication

necessary for Bayeh, because Bayeh suggests turning the prior art for implementation details,
and because it would have enabled the servlets of Bayeh to be reloaded on the fly (p. 30, para. 1).

Regarding dependent claim(s) 9 and 19, Bayeh teaches input to the filter is a XSL style-sheet
that determines its functionality (col. 9, ll. 4-6). As the filter is a servlet object, its input is
passed through a parameter. As the filter processes an XSL stylesheet, equivalent to a
transformation script, it is deemed to be an XSL processor.

Regarding dependent claim(s) 5 and 15, Bayeh teaches selecting a particular servlet based on
whether or not it is busy (col. 8, ll. 43-48). This amounts to a selection scheme that takes into
account conversion time, since pickling a non-busy servlet would decrease conversion time.

Regarding dependent claim(s) 16, Bayeh teaches the filter converts source data from a first
format to a second data format (col. 4, ll. 37-42).

5.   **Claim 20 is rejected under 35 U.S.C. 103(a) as being unpatentable over Bayeh.**

Regarding independent claim(s) 20, Bayeh teaches a partial filter library as part of the server
(col. 7, ll. 36-38). Bayeh teaches retrieving a rule set for a plurality of filters, or servlets, (col. 8,
ll. 36-64), wherein upon being chained the filters convert source data to requested data (col. 4, ll.
23-37). Bayeh discloses that parts of its system are used to determine a source data format (col.
9, line 64 – col. 10, line 1). Bayeh discloses that chain factory is called that has a source data
format and a target (col. 9, ll. 30-63, or col. 9, line 64 – col. 10, line 15). Inherently as the
objects are performing task, they must have been instantiated. Therefore, Bayeh although Bayeh
does not explicitly teach the conversion service, protocol reader, chain factory, service manager
and filter registry service, Bayeh does perform all the functions performed by the conversion
service protocol reader, chain factory, service manager and filter registry service as described

above. As such absent any evidence of the criticality of using these specific objects to accomplish such tasks, it would been obvious to one of ordinary skill in the art at the time of the invention to separate the tasks already performed Bayeh's server into any number of distinct objects, as it would improve readability of code as well as make maintenance and troubleshooting easier.

6.   **Claims 7, 17, 22, and 23 are rejected under 35 U.S.C. 103(a) as being unpatentable over Bayeh and Zeigler as applied to claims 1, 13, and 21 above, and further in view of Garshol, "Free XML Software", (12/15/199).**

Regarding dependent claim(s) 7, 17, and 22, Bayeh does not teach a Simple API for XML. Garshol teaches a Simple API for XML (p. 22), which includes a plurality of filters that can accept the input and output of themselves (p. 24, "Parser Filters"). It would have been obvious to one of ordinary skill in the art at the time of the invention to replace Bayeh's servlets with Garshol's filters, as SAX was a de facto standard at the time of the invention (p. 22, "SAX").

Regarding dependent claim(s) 23, SAX's interface is inherently an XML document handler interface, as proven at least by Applicant's specification (p. 39).

Response to Arguments

7.   Applicant's arguments filed 08/01/2006 have been fully considered but they are not persuasive.

Regarding Applicant's remarks on Pages 4-6 regarding claim 1:

Applicant alleges that the Bayeh does discuss servlet communication, and then rehashes the portions of Bayeh that discuss MIME types and server aliasing. Applicant then appears to claim that since "communication" is not explicitly stated, that Bayeh does not suggest any

communication. However, the communication is implicit in the language of Bayeh. The cited

potions deal with a stream passing through a plurality of filters. They must be in communication

with each other in order to pass that stream. Bayeh explicitly says to look to the prior art to learn

how to do so, and Ziegler specifically teaches a preferred way of communication.

Regarding Applicant's remarks on Pages 7-9, regarding claim 1:

Applicant alleges that Zeiger does not mention different data types. This has been

corrected to read different servlet types.

Applicant alleges that calling a method of another servlet using an interface teaches or

suggests nothing about how it implemented. As one of ordinary skill in the art would recognize,

that is the point of an interface. An interface is specifically left unimplemented so that each

servlet may implement it, as is taught in Zeigler. In fact the example does have bar servlet

implementing that function (p. 31).

Applicant alleges that the interfaces of Ziegler are not generic source and target data

format independent interface. Applicant shows examples of interfaces (for example, Table 8 on

page 29), but provides no clear definition of how "generic source and target data format

independent" modifies the term "interface." Therefore, the term is treated as broadly as the plain

meaning requires. As the interface is for servlets, which do not care about the underlying data

format, merely that there is a text stream, the interface of Zeigler is not seen to be any different

then the interface of the Applicant. Therefore it is a generic source and target data format

independent interface, in accordance with the terms use in the specification.

In response to applicant's argument that there is no teaching of how the interfaces would

have been incorporated, the test for obviousness is not whether the features of a secondary

reference may be bodily incorporated into the structure of the primary reference; nor is it that the

claimed invention must be expressly suggested in any one or all of the references. Rather, the

test is what the combined teachings of the references would have suggested to those of ordinary

skill in the art. See *In re Keller*, 642 F.2d 413, 208 USPQ 871 (CCPA 1981).

Applicant alleges that techniques for triggering chained processing are at a different

functional level then call a method in another servlet. Applicant offers not further explanation

and as such this amounts to a mere allegation.

Applicant alleges that the motivation is unrelated because it deals with chained

processing instead of communication. As explained above, chained processing involves

communication because the servlets must pass the data stream to each other.

Applicant alleges there is no teaching of the interface belonging to the data servlet and

the rendering servlet. However, that is only one embodiment of Bayeh. The portions involving

chaining deal with generic servlets not necessarily the data or rendering servlet as exemplified by

84'.

Regarding Applicant's remarks on pp. 10-11, regarding claims 3 and 20:

Applicant alleges that Quirksmode does not teach what is stated. Specific citations have

been added. In response to the argument that Quirksmode appears to contradict Bayeh, the

Office submits that Bayeh is merely oversimplifying browser technology. Quirksmode is

evidence of the complete range of capabilities. Additionally, anyone who has used a browser, in

addition to skilled artisans, recognizes that browsers can process images, which are a different

data format.

Regarding Applicant's remarks on pp. 11-12 regarding claim 20:

Applicant alleges that limitations have been ignored. The Office has rephrased the rejection to make it clear that the limitation have not been ignored.

Conclusion

8.   **THIS ACTION IS MADE FINAL.** Applicant is reminded of the extension of time policy as set forth in 37 CFR 1.136(a).

A shortened statutory period for reply to this final action is set to expire THREE MONTHS from the mailing date of this action. In the event a first reply is filed within TWO MONTHS of the mailing date of this final action and the advisory action is not mailed until after the end of the THREE-MONTH shortened statutory period, then the shortened statutory period will expire on the date the advisory action is mailed, and any extension fee pursuant to 37 CFR 1.136(a) will be calculated from the mailing date of the advisory action. In no event, however, will the statutory period for reply expire later than SIX MONTHS from the mailing date of this final action.

9.   Any inquiry concerning this communication or earlier communications from the examiner should be directed to Adam M. Queler whose telephone number is (571) 272-4140. The examiner can normally be reached on Monday-Friday.

If attempts to reach the examiner by telephone are unsuccessful, the examiner's supervisor, Steven Hong can be reached on (571) 272-4124. The fax phone number for the organization where this application or proceeding is assigned is 571-273-8300.

Information regarding the status of an application may be obtained from the Patent

Application Information Retrieval (PAIR) system. Status information for published applications

may be obtained from either Private PAIR or Public PAIR. Status information for unpublished

applications is available through Private PAIR only. For more information about the PAIR

system, see http://pair-direct.uspto.gov. Should you have questions on access to the Private PAIR

system, contact the Electronic Business Center (EBC) at 866-217-9197 (toll-free). If you would

like assistance from a USPTO Customer Service Representative or access to the automated

information system, call 800-786-9199 (IN USA OR CANADA) or 571-272-1000.

aq

STEPHEN HONG
SUPERVISORY PATENT EXAMINER